



An experiment-driven energy consumption model for virtual machine management systems

Mar Callau-Zori, Lavinia Samoila, Anne-Cécile Orgerie, Guillaume Pierre

► To cite this version:

Mar Callau-Zori, Lavinia Samoila, Anne-Cécile Orgerie, Guillaume Pierre. An experiment-driven energy consumption model for virtual machine management systems. *Sustainable Computing: Informatics and Systems*, 2018, 18, pp.163-174. 10.1016/j.suscom.2017.11.001 . hal-01632962

HAL Id: hal-01632962

<https://hal.science/hal-01632962>

Submitted on 10 Nov 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An experiment-driven energy consumption model for virtual machine management systems

Mar Callau-Zori^a, Lavinia Samoila^a, Anne-Cécile Orgerie^{a,*}, Guillaume Pierre^a

^aUniv Rennes, Inria, CNRS, IRISA, Rennes, France

Email: {mar.callau-zori, anne-cecile.orgerie, lavinia.samoila, guillaume.pierre}@irisa.fr

Abstract

As energy consumption is becoming critical in Cloud data centers, Cloud providers are adopting energy-efficient virtual machines management systems. These systems essentially rely on “what-if” analysis to determine what the consequence of their actions would be and to choose the best one according to a number of metrics. However, modeling energy consumption of simple operations such as starting a new VM or live-migrating is complicated by the fact that multiple phenomena occur. It is therefore important to identify which factors influence energy consumption before proposing any new model. We claim in this paper that one critical parameter is the host configuration, characterized by the number of VMs it is currently executing. Based on this observation, we present an energy model that provides energy estimation associated with VM management operations, such as VMs placement, VM start up and VM migration. The average relative estimation error is lower than 10% using the transactional web benchmark TPC-W, making it a good candidate for driving the actions of future energy-aware cloud management systems.

Keywords: Green computing, Cloud computing, Virtual Machine Management, Energy model, Virtual Machine migration.

1. Introduction

Advances in distributed systems have historically been related to improving their performance, scalability and quality of service. However, the energy consumption of these systems is becoming more and more concerning. Although the emergence of Cloud Computing has led to a massive virtualization of the resources, their energy cost is still real and rapidly increasing due to a growing demand for Cloud services. As an example, for 2010, Google used 900,000 servers which consumed 1.9 billion kWh of electricity [1]. This ever-increasing electricity bill puts a strain on operating expanses, and has thus led researchers to look for more energy-efficient frameworks.

*Corresponding Author: Anne-Cécile Orgerie

The most common approach to save energy in Cloud infrastructures consists in consolidating the load on the smallest possible number of physical resources [2], thus making it possible to switch idle machines to sleep modes [3]. To this end, scheduling heuristics are used for allocating new virtual machines (VMs) or for re-allocating running VMs in conjunction with live-migration techniques for dynamic consolidation, with a close-to-zero downtime.

In this paper, we propose and evaluate an experiment-driven model to estimate the energy consumption associated to VM management operations, such as VM placement, VM start up and VM migration. This model provides reliable and accurate energy consumption values, which are required by energy-aware VM management frameworks in order to take efficient decisions.

1.1. Motivating example

Cloud providers offer on-demand computational resources in the form of virtual machines, disks and network functions. These resources can be used freely by the tenants. Meanwhile, the provider may perform VM management operations, such as create/delete, suspend/resume, shut down/restart and migrate in order to reduce the energy consumption of the platform or simply for maintenance purposes or to redistribute the load. These operations can be performed in different ways leading to different energy consumption.

A scenario based on consolidation illustrates the variability in energy consumption of two typical algorithms in Figure 1. The consolidation goal is to reduce the number of physical hosts where VMs are allocated. The example starts with four identical hosts (same hardware) and eleven identical VMs uniformly distributed among the hosts (left-side of Figure 1). The final configuration in both cases has only two provisioned hosts – allowing to switch off the two others.

Figure 1 on the bottom shows the consumption for the two algorithms: the *first-fit consolidation* where all VMs are moved to host 1, and the *balancing consolidation* where VMs are uniformly distributed between host 1 and host 2. Our experimental results provide the energy overhead due to the migrations and the power consumption of the hosts after the re-configuration. Despite the numerous approaches based on the first-fit algorithm in the literature [2], the balancing consolidation performs better on this example. Indeed, it saves 12% of the energy with respects to the first-fit consolidation for the migration operations, and the final consumption of the two hosts is less by 4.5% with this configuration. In particular, this example clearly highlights that identical VMs are not equal in terms of energy consumption.

This paper argues that VM management systems must be aware of the energy consumption of different operations (such as VM allocation, VM start up and VM migration) and must have an accurate energy model of their physical hosts depending on their configuration (i.e. number of hosted VMs) in order to be truly energy-efficient. More details about these experiments are provided in Section 6.

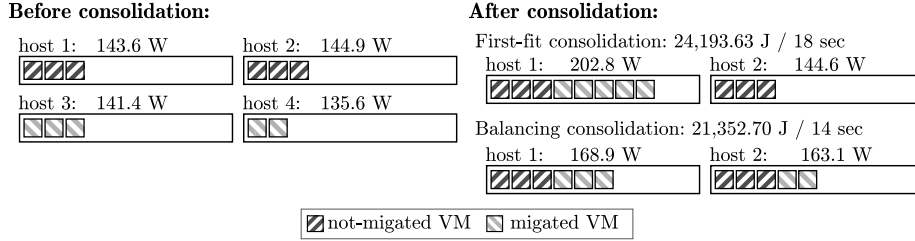


Figure 1: Energy overhead depends on the consolidation algorithm.

1.2. Experimental evaluation

This paper targets online applications hosted by web servers on Cloud infrastructures and represented here by the TPC-W benchmark [4] for evaluation purposes (Section 5). TPC-W is a transactional web benchmark simulating a Web-based shop for searching, browsing and ordering books. We have validated the proposed energy model through two use-case scenarios using OpenStack as the Infrastructure-as-a-Service Cloud layer. In the first scenario, the system has a workload peak which produces an increasing number of VMs allocated on the physical host. In the second scenario, migrations and VMs starts-up occur in a system with several hosts. In both cases, the model is able to predict the involved energy consumption with an average relative error lower than 10%, and thus to adapt the system accordingly.

The applicability of the model is studied over a realistic scenario (Section 6). In particular, we study overcommit situations where a server hosts more VMs than it should with respect to its physical capacities and the virtual resource specifications of the hosted VMs. Although this behavior is not often studied in literature, it is frequent on Cloud infrastructures [5, 6]. The model is trained over a simple scenario and then applied to a more complex scenario on which it is used to evaluate the energy-efficiency of three consolidation algorithms. We hereby demonstrate the ability of the model to provide valuable results for a what-if analysis. Experiments on a real platform exhibit an average relative error lower than 10%. The advantages and drawbacks of our model are further discussed in Section 7.

1.3. Contributions

In summary, this paper makes the following key contributions: (1) we provide an experimental study of the energy consumption of VM management operations under realistic workload conditions including overcommit situations, (2) we propose an energy model for physical host consumption, (3) we evaluate the accuracy of this model via experimentation on a real system deploying web applications, and (4) we show its applicability and how it can significantly help Cloud management systems to take truly energy-efficient decisions.

The rest of the paper is organized as follows. Section 2 introduces the related work. Experimental measurements are presented in Section 3. Section 4

describes the model for energy consumption estimation. Section 5 presents results in model estimation. The model calibration methodology is described in Section 6. Section 7 discusses the present work, and Section 8 concludes and presents future directions.

2. Related work

Energy models have mainly been studied in the context of a single host, a single virtual machine or an entire cluster [7]. The virtualization proposes to improve resource utilization through dividing one physical server into multiple isolated virtual machines. Resources sharing makes complex to leverage experiences from previous efforts on energy estimation in single host [8, 9, 10, 11, 12, 13] and on energy estimation in cluster models for data centers [14, 15].

In virtualized environments, we distinguish previous models to estimate power consumption of a single VM and previous models to estimate power overhead during live-migration. In the rest of this section, we discuss research in both directions.

2.1. *Energy-efficient VM management techniques*

On the one hand, heuristic-based methods for VM allocation take practical decisions to satisfy immediate goals, such as maximizing the number of idle hosts [16, 17, 18], minimizing the number of VM migrations [19], or determining the most energy-efficient available host for each new VM [20, 21]. However, solutions derived from heuristics are not guaranteed to be optimal in the long term, especially in heterogeneous environments [22]. Therefore, it is unclear which heuristic will guarantee the most energy-efficient system. On the other hand, although live-migration leads to almost no performance degradation, it is not energy-free. While for the hosts, the migration cost mainly depend on the VM size and network bandwidth [23, 24], the sequential aspect of multiple VMs migration has also an impact [25], and widely-used Cloud software platforms, like OpenStack [26] for instance, perform multiple migrations in a sequential way. These points enlighten the necessity of accurate energy models of VM management operations for applying the adequate energy-efficient technique.

However, defining an accurate energy consumption model for VMs is still an open challenge [22]. This is mainly because the energy consumption depends on multiple factors: the workload, the hardware, the host configuration, the VM characteristics, etc. Furthermore, determining the suitable parameters which describe the inherent properties is crucial to enforce the model. Previous work [27, 28, 29, 30, 31, 32] has focused on models based on factors, such as the server hardware, the Cloud software stack (such as the hypervisor and the management software) or the VMs resources (such as CPU, memory, disc and cache). However, the impact of the host configuration has been slightly studied.

2.2. Energy model for VM management operations

This paper presents directions for a comprehensive energy-estimation tool assuming that the VM management system performs actions sequentially, as it is often the case in current Cloud software platforms [26]. This means that if the system receives two requests (e.g. a VM creation and a VM migration), it waits for the first one to finish before executing the other one. Our model is built over a wide set of experiments aiming at observing variations according to VMs management operations, host configuration and workload application. This experimental study (Section 3) provides helpful information claiming for an energy model based on VMs management operations and hosts configuration. The proposed model (Section 4) takes into account the host configuration, and in particular the number of already running VMs on this host.

In [33], the authors study the energy consumption according to the number of VMs allocated in a single host and to the number of virtual CPU. Similarly to our motivating example (Section 1.1), they show that the energy consumption depends on the previous configuration of the host. However, in their experiments, the number of virtual cores is never greater than the number of physical cores. So, they do not explore overcommit situations, which are frequent in Cloud platforms [5, 6], although not studied in literature from an energy point of view. In this paper, we explore host configurations with CPU overcommit, and we show that the proposed model works for these overcommit situations.

2.3. VM power models

In [31], the authors propose a linear power model in terms of CPU utilization, memory (last level cache misses) and disk, with a linear regression to learn model parameters. Authors achieve a total power error on the host of 1.6 Watts to 4.8 Watts.

Another linear model is presented by Bohra and Chaudhary in [27] considering CPU, memory, disk and cache. The principal component analysis obtains a high correlation between CPU and cache; and between disk and DRAM. This fact motivates the utilization of two regression models to estimate model parameters. The results show that the model is able to predict energy power with an average mean error of 7%.

In [32], Krishnan et al. present a linear model considering CPU and memory. They indicate the importance in energy consumption of the utilization of the cache levels and memory level parallelism. The evaluation shows that VM power is estimated within 7% of the measured value.

Gaussian mixture models are proposed in [30] to estimate power consumption based on architecture metrics (such as IPC, cache, memory usage, among other performance monitored counters). In the training phase, a set of Gaussian mixture models that represent different architecture interactions and power levels are obtained. An average prediction error of less than 10% is achieved.

Another linear model based on CPU utilization, memory and disk is proposed in [28]. This model is learned according to the dynamic voltage frequency

(using the DVFS technique). For the evaluation, up to 10 virtual CPUs are allocated in servers with 2 cores. The results exhibit an average error lower than 5%.

Based on the major power consumer which is the CPU for a server, a process-level power estimation is presented in [29]. A polynomial regression is applied to estimate the power consumed by the CPU in terms of the frequency and the unhalted cycles for each active process and core. The average estimation error achieved is below 10%.

In the previous work, we observe three main characteristics. Firstly, presented models assume that a VM always consumes the same amount of energy if the VM performs similar work (e.g. similar utilization of CPU, memory and cache). Thus, these contributions assume that energy consumption of a VM is independent of the number of hosted VMs in the hosts. Secondly, the total number of virtual cores used by the running VMs is at most the number of physical cores (except in [28]). However, in real Cloud platforms, over-provisioning is common for users, leading Cloud providers to resort to over-commitment [5, 6]. Hosting more virtual cores than the number of physical ones can consequently affect the assumption that energy consumption of a VM is independent of the number of hosted VMs on the server. Thirdly, most of the previous work is evaluated through benchmarks specifically designed to stress CPU, memory and/or I/O. However, in real environments, resources do not reach such high utilization levels, mainly to guarantee performance and quality of service. Only work presented in [29] employs for their evaluation SPEC JBB 2013 [34], a benchmark which emulates a supermarket company.

2.4. Live-migration power models

Previous analysis about energy overhead during live-migration shows differences in the energy consumption for source and destination hosts [35, 11, 36].

In [37], a model based on the VM memory size, the memory transmission rate and the memory dirtying rate is proposed. The authors not only estimate the energy overhead during migration, but also the network traffic overhead and the downtime. However, they assume that energy consumption is similar in source and destination hosts. The evaluation using Xen hypervisor [38] shows an average error below 10%.

It is shown in [23] that energy overhead during migration depends on RAM and available bandwidth. Based on this work, the authors propose in [39] a power model for estimating the energy overhead of live-migration for an idle VM. The average error using KVM hypervisor [40] is lower than 10%.

In [41], the authors present a model depending on the CPU utilization (considering both hosts and the migrated VM), the VM memory utilization, the available bandwidth and the memory dirtying rate. The evaluation is conducted through live-migrating a VM in a host where there are already 8 running VMs. Results using Xen hypervisor with a CPU and memory intensive workload shows a normalized root mean square error lower than 18%. Authors expended this work in [42] by proposing a model for network transfers achieving a normalized root mean square error lower than 9%.

One can notice that, in the previous work, only one VM is migrated for the validation of the various models. However, the consolidation usually implies de-allocation of all the VMs of a host. Therefore, migrations are produced consecutively and energy overhead can be affected due to these successive migrations.

3. Experimental study

We conduct an in-depth experimental analysis of VM management operations in order to understand their impact on energy consumption. The experiments have been executed on Grid’5000 [43], a French test-bed designed for experiment-driven research. We base experiments on four servers equipped with two 6-cores Xeon E5-2630@2.3GHz processors (12 cores in total), 32 GB of RAM, 10 Gbps Ethernet and embedding with a Nvidia Tesla M2075 GPU accelerator. The power consumption is collected externally at each second for each server using watt-meters provided by the SME Omegawatt¹ specially for the Grid’5000 platform. We use OpenStack Icehouse [26] to dynamically create VMs through the KVM hypervisor [40]. VM deployment is relying on an NFS server (Network File System) for the storage. This solution is well suited for web servers, and has a facilitating impact on VM live-migration. The VMs are configured with 4096 MB memory, 80 GB of virtual disk and 4 virtual CPUs. This configuration is similar to Amazon EC2’s *m4.xlarge* instance type (4 vCPU, 16 GB of memory), although with less memory [44]. As the hosts considered in this experimental study comprise 12 cores each, without overcommit, they can only run three of such VMs each.

We create realistic workload by running the MySQL+Java version of TPC-W in the VMs, as this version is representative of current Cloud applications. TPC-W is a transactional web benchmark which simulates the activities of a business oriented transactional web server [4]. TPC-W allows three profiles of web traffic (called *ordering*, *browsing* and *shopping*). The main traffic profile used is *ordering*, where the number of book’s purchases is large – this means, large number of write operations into disk. Other TPC-W parameters are summarized in Table 1. TPC-W measures the throughput in number of WIPS (Web Interactions Per Second) that the server manages to sustain. This metric is employed to characterize the application performance.

Summarizing, each VM has the virtual hardware described above and runs a TPC-W server. We therefore measure two metrics: the power consumption of each physical server, and the throughput measured in WIPS of each TPC-W server, each one running in a single VM. Each experiment was run on the four different servers. Following figures display the average measurement results performed on these four servers.

We study three kinds of VM management operations: the VM allocation in a single host (Section 3.1), the VM start up (Section 3.2) and the VM migration

¹<http://www.omegawatt.fr/gb/index.php>

Table 1: TPC-W configuration parameters.

Number of browsers	100
Transaction Mix	Factory
Ramp-up time	600 sec
Ramp-down time	600 sec
Measurement time	1800 sec
Slow-down factor	1.0 (default)
Nb. customers	144000
Nb. items	10000
Think time multiplication	0.25
Maximum errors allowed	unbounded
Maximum number of images down-loaded at once	4

(Section 3.3) Then, we study the impact of the workload profile in Section 3.4 and the influence of the GPU accelerator in Section 3.5.

3.1. Impact of the number of VMs allocated in a host

Our first experiment evaluates how many VMs a single host can manage in an energy-efficient way. The scenario consists in increasing the number of VMs put on a single host. Figure 2 shows the power consumption and throughput depending on the number of VMs running on the host. The X-axis shows the number of VMs in the host, whereas the Y-axes present three metrics: the power consumption in Watts (blue circles, left side), the total throughput over all VMs in WIPS (green triangles, right side), and the power/performance ratio in Joules/WI (Joules over number of web interactions, red squares).

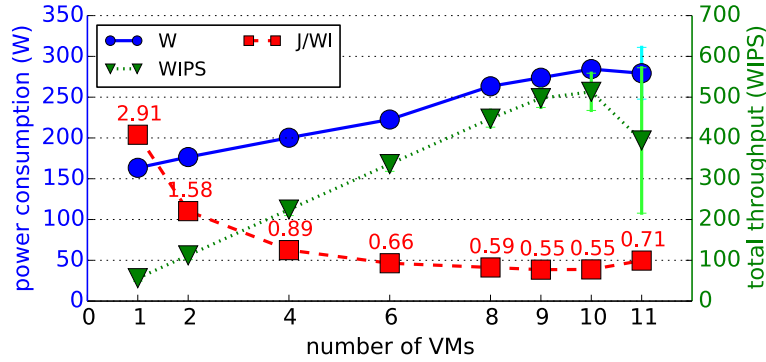


Figure 2: Power consumption and throughput when varying the number of VMs in the host.

The figures show that the power consumption and VM throughput increase up to 11 VMs per host, but increasing is linear up to 8 VM. If we keep increasing

Table 2: Average power consumption per VM.

Nb. of VMs	1	2	4	6	8	9	10
Power per VM (W)	67.5	40.3	26.1	21.1	20.9	19.8	18.9

the number of VMs, then the consumed power starts decreasing from 11 VMs and throughput also drops at 11 VMs. The optimal power/throughput ratio is obtained using 9-10 VMs per host.

If focusing only on throughput, the most efficient configuration is the one ensuring linear scalability: so 8 VMs per host at maximum. However, maintaining a maximum of 8 VMs per host is not the most power-efficient case. For example, a system with 1 host with 10 VMs spends less power (219.84 W) than a system with 2 hosts: one with 8 VMs and another with 2 VMs, (335.76 W) saving 52.73% of power. This relationship between power and throughput corresponds to the red line in Figure 2, and shows that placing either 9 or 10 VMs per host is more power-efficient than keeping 8.

Regarding power consumption in details, Figure 3 presents a synthetic view of the power consumption distribution using box-plot. Classically, the box-plot has a box and a whisker plot for each case. The box extends from the 25th to 75th percentile with a red line at the median. The whiskers extends from the box, and points outside the whiskers are considered outliers. Distances from whiskers to box are a factor of 1.5 of the interquartile range (box height). Finally, outliers are plotted individually.

In the power distribution, Pearson correlation² coefficient (0.97) with the number of VMs in $[1, 10]$ shows a strong linear correlation between the number of VMs and the power consumption. A special case occurs for 6 VMs: where the 75th percentile (top on the box) is lower than the regression line. A dotted line is employed for the end of the linear regression because of the particular unstable behavior of servers when heavily over-committed.

As we said, previous work assumes that the power consumption is a linear function in the number of VMs. Concretely, the assumption on the power of the host is $P_{\text{host}} = P_{\text{idle}} + \sum_{v \in \text{VMs}} P_v$, where P_{idle} is the idle power of the host and P_v is the power of the VM v . In our case, the idle power is 95.92W (measured independently) and all the VMs perform the same work having the same consumption before saturation (less than 9 VMs). However, in Table 2, we show the average power value for one VM, P_v , and hereby evidencing that the power consumed in one VM depends on the number of running VMs on the host.

From this first experiment, we can draw several conclusions. Firstly, in order to achieve energy-efficiency, allowing small degradation of the throughput can generate important power savings, as it has already been noticed in [22].

²<http://mathworld.wolfram.com/CorrelationCoefficient.html>

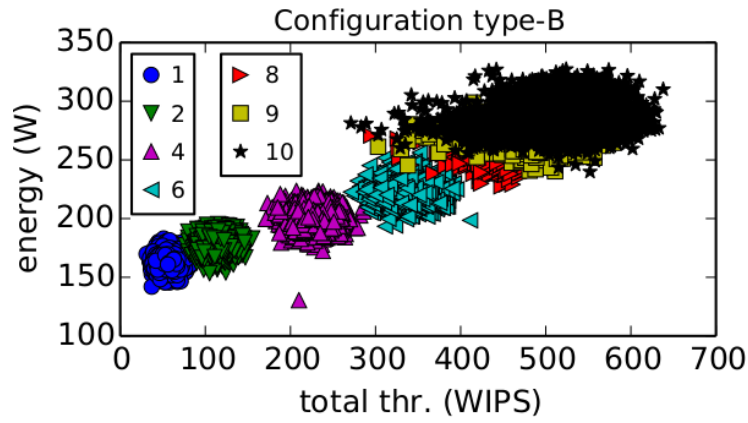
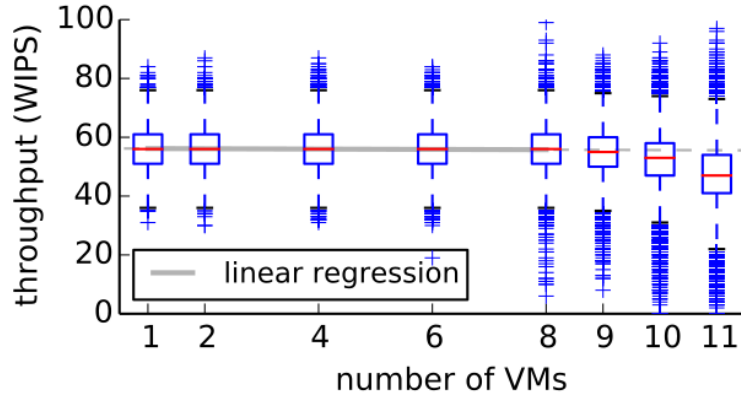
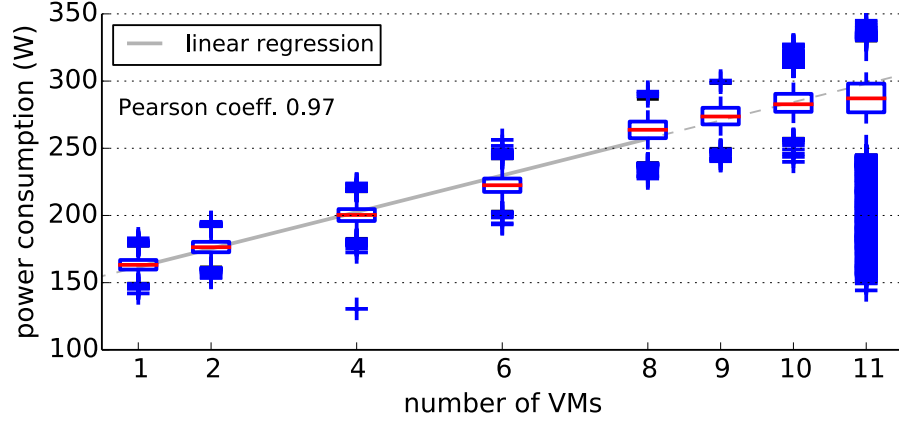


Figure 3: Details on power consumption by increasing the number of VMs in the host.

Secondly, power consumption is significantly impacted by the host configuration (i.e. the number of running VMs).

3.2. Impact of VM start up

We now investigate the power consumption overhead and throughput degradation during a VM creation. Figure 4 shows an example of VM creation on a host already running one VM. After 1200 seconds, a new VM is created. VM start-up finishes when TPC-W starts to run (in this example it takes 28 seconds). As expected, the throughput does not increase during the creation. However, starting a new VM causes a spike in power consumption (with consumption peak greater than 180W, representing an increase of about 15% compared to the original consumption).

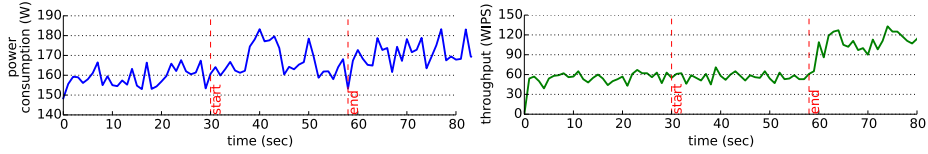


Figure 4: Example of VM start-up: power consumption and host throughput.

After this example, we provide a general study in Figure 5 representing average power consumption and throughput variation during a VM creation. The left side (blue colors) corresponds to the total power consumption, while the right side (green colors) represents the average throughput (WIPS) across all the VMs. Power consumption bars (respectively throughput bars) are split in two: first, the power consumption (respectively throughput) before VM start up; and second, the power consumption (respectively throughput) observed during VM start-up (averaged over the start-up duration).

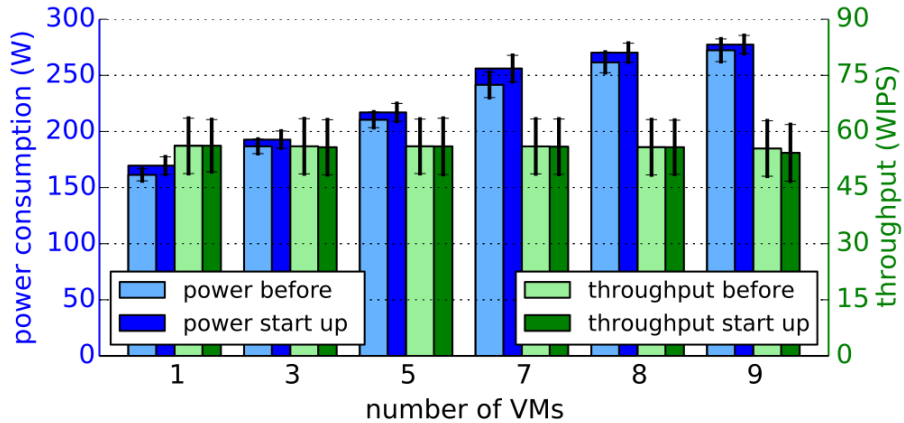


Figure 5: Throughput and power consumption overheads upon VM start-up.

Table 3: Average power consumption overheads upon VM start-up.

Nb. of VMs	1	3	5	7	8	9
Power overhead	5.1%	3.2%	3.1%	6.0%	3.3%	1.9%

While the power consumption is always greater during start-up, the throughput does not suffer degradation except for 9 VMs. One can notice that degradation happens passing from 9 to 10 VMs, which confirms the previous conclusion asserting that the 10 VMs case breaks the linear scalability.

Details about power consumption during start-up process are presented in Figure 6. The gray line represents the linear regression over data obtained from 1 to 8 VMs before the VM start-up. We see that power increases during start-up from 1 to 8 VMs. Moreover, power consumption overhead is smaller for 6 VMs than for the other cases.

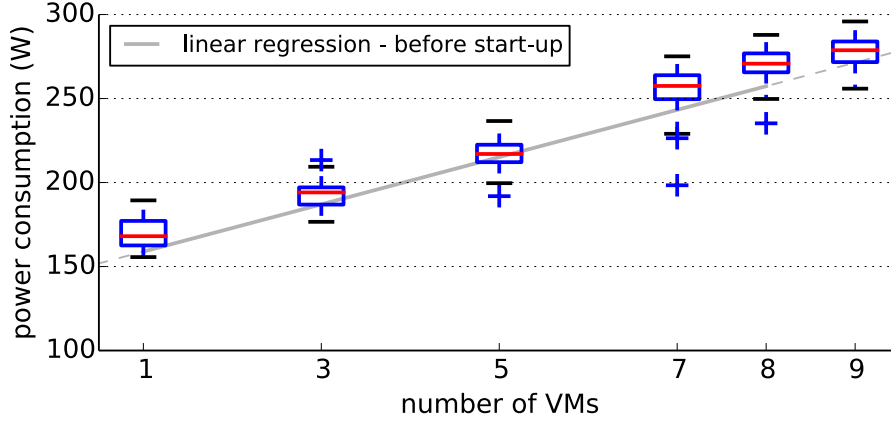


Figure 6: Details in power consumption overhead upon VM start-up.

We have explored this phenomenon more deeply in Table 3, representing the power overhead percentage of VM start-up. We observe a decreasing trend in the power overhead of 1-5 and 7-9.

We clearly see that starting up a VM is not power-neutral. Hence an energy-efficient system should guarantee a minimum lifetime after VM creation. Furthermore, this minimal lifetime depends on the previous configuration on the host.

3.3. Impact of VM migration

We now evaluate the impact on power consumption and throughput of VM live-migration. The scenario consists of migrating a VM between two hosts which already contain some running VMs. Under the live-migration with pre-copy approach [45], the VM continuously runs while its memory is copied from

source host to destination host in several runs (pre-copy phase). Requests are processed by the VM in the source host, until the system is notified of the pre-copy phase termination. At this point, the VM is suspended to perform the last copy (stop-and-copy phase). Finally, requests start to be processed by the VM in the new location.

Figure 7 presents the experimental results of a VM migration in terms of power (top) and throughput (bottom). Throughput is equal to zero during few seconds, waiting for the system to be informed of the new location. Power consumption overhead is investigated in the source and in the destination hosts separately.

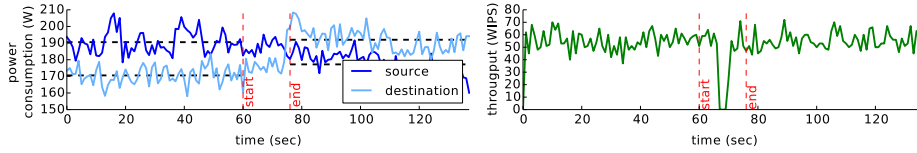


Figure 7: Example of power consumption and throughput upon VM migration.

During migration, the power consumption of source host starts to decrease, while in the destination host, it increases. Dotted lines represent the average power consumption in source and destination hosts, before and after migration. Interestingly, it takes about 20 seconds after the end of the migration for both hosts to reach power consumption at levels corresponding to the number of allocated VMs. We believe that these 20 seconds are due to the VM post-copy algorithm. Hence, the experiment show that live-migration produces power consumption overhead beyond the migration time.

For a deeper understanding of the impact of migration, we present a general study where a VM is migrated after 1200 seconds. Source and destination hosts already have a number of static (i.e. non-migrated) VMs. This means that we distinguish three types of VMs: migrated VM (*mig*), static VMs in the source host (*st-src*) and static VMs in the destination host (*st-dst*). In the following, for the sake of clarity, we take the same number of static VMs in source and destination host for a given experiment ($st-src = st-dst$).

Results are summarized in Figure 8. The X-axis represents the number of static VMs for each host (source and destination). Fixing the number of static VMs, we have the power consumption of the source and destination host: in lighter blue the power consumed before migration and in darker blue the power consumed during migration. On the throughput side, we consider three metrics: one for the migrated VM, one for the static VMs in the source host, and, respectively, in the destination host.

During migration, the power consumption of the destination host increases (1.35%, 7.63%), and in the source host, it does not have a stable behavior either (-2.96%, +0.92%).

Details about the total power consumption overhead are provided in Figure 9. The total power is the accumulated power consumed in the source and destination hosts during migration. The gray line represents the regression line

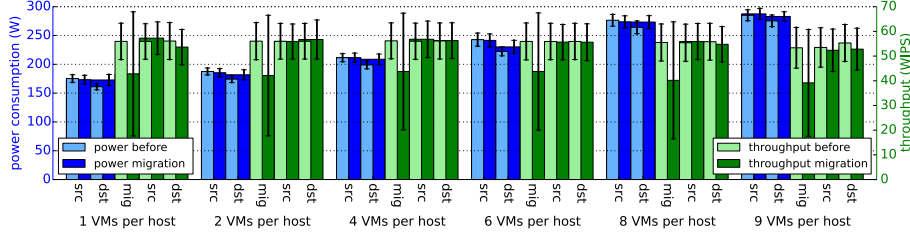


Figure 8: Overheads upon VM migration.

over the total power consumption before the migration. The power increment is more significant in extreme cases (VMs per host = 1, 2, 8 and 9), while for 4 to 6 VMs per host, power does not significantly increase during migration. As expected, the total power consumption increases during migration (0.55%, 2.69%).

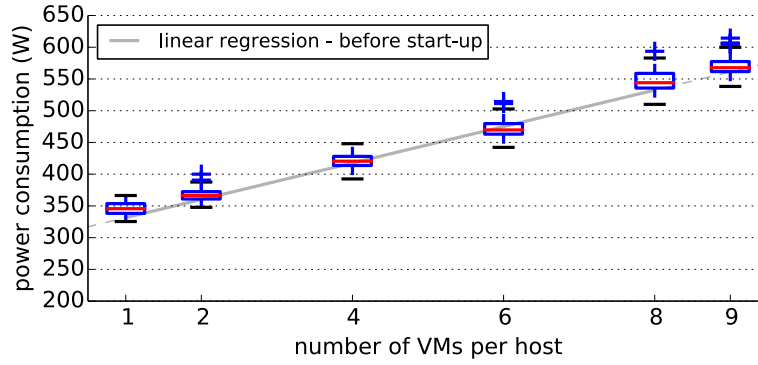


Figure 9: Details in power consumption overhead upon VM migration.

As explained in the example from Figure 7, it takes time for the power consumption in the destination host after migration to stabilize. To explore this issue, we determine the minimum time after a migration where the average power consumption in the destination host, on a window of 10 seconds, is less or equal than the percentile 60% of the power consumption in source host before the migration. The 60% threshold has been determined empirically to highlight the time when a migration has finished. We are exploiting the fact that after migration, the destination host allocates the same number of VMs than the source host before the migration. These times are summarized in Table 4. We observe a non-negligible impact on the power consumption: the time to recover expected power consumption varies between 1 and 271 seconds depending on the host configuration.

The throughput of non-migrated VMs is not affected by the migration, as shown in Figure 8. However, the migrated VM has an important degradation reducing its throughput by 21% to 29%. Although migration takes only a few

Table 4: Time to recover expected power consumption after a VM migration.

Nb. of VMs	1	2	4	6	8	9
Delay (sec)	1.3	14.8	36.5	97.0	61.8	271.5

seconds, the impact of the resulting throughput degradation will largely depend on the moment it happens. The average down-time varies from 2.73 to 3.30 seconds being similar to that obtained in the literature [46]. We expected the downtime to be independent of the hosts configuration because hosts are not saturated. However, our experimental study shows that it is not the case.

As other studies [11] have already pointed, live-migration produces energy overhead occurring mainly in the destination host. Moreover, we have also observed that migration produces power consumption overhead also after the migration time. This fact is a sensitive issue in consolidation, because migrations are often produced consecutively. Finally, we see that the power consumption overhead depends on the number of running VMs in the hosts.

3.4. Impact of workload profile

We now evaluate the impact of the client behavior as it is influencing the workload on the servers. As previously stated, TPC-W allows to simulate three different profiles by varying the ratio of browse to buy transactions: shopping, browsing and ordering.

Figure 10 shows a scenario where the number of VMs put on the same host increases for the TPC-W profiles *ordering* and *browsing*. While the X-axis represents the number of VMs in the same host, the Y-axes represent two metrics: the average power consumption in Watts (blue colors, left side) and the average throughput in WIPS (green colors, right side).

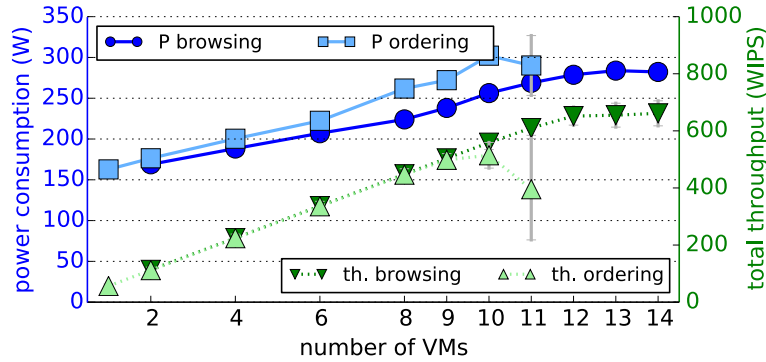


Figure 10: Power consumption and throughput according to workload profiles.

We observe that throughput is equal for the two profiles between 1 and 9 VMs. From 10 VMs however, *ordering* workload starts to degrade, reaching

saturation at 11 VMs. On the other hand, placing fewer than 13 VMs with *browsing* workload has no negative impact on the throughput. This behavior – faster host saturation with *ordering* workload than with *browsing* workload – was expected, because for *ordering*, more writes to the disk are performed.

In terms of power consumption, it is larger under *ordering* than under *browsing*, while both have the same throughput. The difference is greater as the number of VMs is increasing. Moreover, under *ordering* the power consumption slope changes at 6 VMs, while under *browsing* the slope break happens at 8 VMs.

3.5. Impact of GPU accelerator

In the previous experiment, all the servers are equipped with an Nvidia Tesla M2075 GPU accelerator. GPU accelerator is designed to use a graphics processing unit (GPU) with a CPU in order to accelerate scientific, analytics, engineering, consumer, and enterprise applications [47].

We investigate the impact of having a GPU accelerator embedded on servers – although we are not using it. Figure 11 presents the results increasing the number of VMs in the host for servers with the GPU accelerator and servers without it (same hardware configuration). As in the previous figures, the X-axis represents the number of VMs in the same host, the Y-axes represent two metrics: the average power consumption in Watts (blue, left side) and the average throughput in WIPS (green, right side).

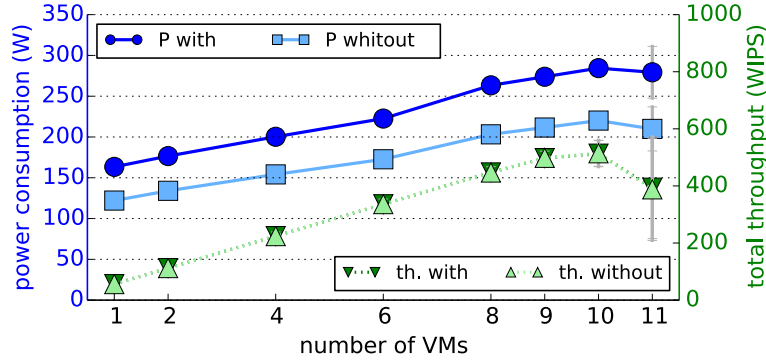


Figure 11: Power consumption and throughput according to GPU accelerator.

As expected, having an unused GPU accelerator does not improve performance. However, even if the GPU accelerator is not used, the power consumption of servers with GPU accelerator is 22% larger than without it. This scenario highlights that power is wasted if unsuitable servers are used to allocate VMs that do not exploit all the hardware capabilities.

Table 5: Model parameters in a host h running m VMs.

VM operation	Model parameter	
	Variable	Description
-	$P(m)$	Power consumption of a host with m VMs (W)
start up	$E_{\text{start}}(m)$	Energy consumption (J)
	$T_{\text{start}}(m)$	Time duration (sec)
migration	$E_{\text{src}}(m)$	Energy consumption in source (J)
	$E_{\text{dst}}(m)$	Energy consumption in destination (J)
	$T_{\text{mig}}(m)$	Time duration (sec)

4. Modeling energy

Next to the experimental facts studied in the previous section, we formalize an energy model for the VMs management operations. The model is additive, this means that the energy consumption of each VM operation (start up and migration) is added to achieve the total energy consumption for a time period.

At every time t , the system can start a VM management operation. The energy consumption of the hosts depends on this choice. Lets $E(t, h)$ the *reconfiguration energy cost* in the host h , that is the energy overhead during the whole VM operation concerning host h ; and t the *reconfiguration time* meaning the time to apply the changes. Noting that while start up and migration imply energy overhead during a interval time; if the systems keeps the same VM placement, the energy consumption corresponds just to one second (assuming we need to provide energy consumption each second).

From results of Section 3, we know that the reconfiguration energy cost depends on the previous state of the host. Therefore, in Table 5, we define the model parameters according to the number of running VMs and the associated VM operation .

Given m the number of VMs in a host h previous to the reconfiguration, the reconfiguration energy cost takes the following values:

$$\left\{ \begin{array}{l} \text{Neither VM start up nor migration begins at time } t \text{ in } h: \\ \quad E(t, h) = 0 \\ \text{A VM start up begins at time } t \text{ in host } h: \\ \quad E(t, h) = E_{\text{start}}(m) \\ \text{A VM migration begins at time } t \text{ implying host } h: \\ \quad E(t, h) = E_{\text{src}}(m) \quad \text{if } h \text{ is the source host and has} \\ \quad \quad \quad \quad \quad m + 1 \text{ VMs at } t \\ \quad E(t, h) = E_{\text{dst}}(m) \quad \text{if } h \text{ is the destination host and} \\ \quad \quad \quad \quad \quad \text{has } m \text{ VMs at } t \end{array} \right.$$

An illustrative example is provided on Figure 12 for the VM start up case. The model has to learn the different parameters, namely $P(m)$, $P(m+1)$, $E_{\text{start}}(m)$ and $T_{\text{start}}(m)$.

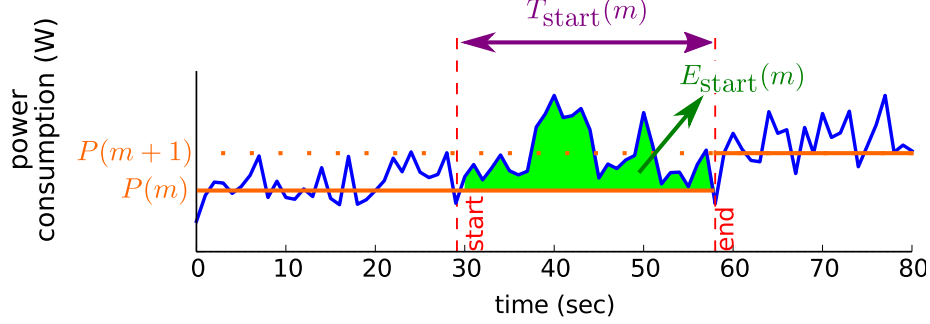


Figure 12: Power consumption when starting a VM.

Finally, the energy consumption of a host during a time period I (during which the number of VMs does not change) is the sum of reconfiguration energy costs during this period plus its regular energy consumed by its m VMs:

$$\sum_{t \in I} E(t, h) + P(m) \times I$$

Parameters described in Table 5 are learned from a set of experiences by linear regression. In the following two sections, we present two realistic scenarios, and we validate our model in both cases. In Section 5, model parameters are learned from the experimental study previously presented. However, in a real environment, it is rarely feasible to perform the experimental scenarios of Section 3 to this end. So, in Section 6, we learn the model parameters from a single scenario on a running system in order to provide a practical method for instantiating our model in a realistic environment.

5. Experimental results about model validation

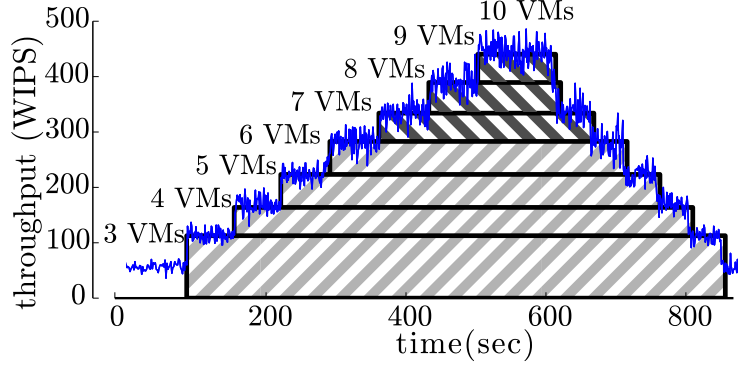
In this section, we validate the model by a set of experiments. These experiments are also designed to show potential use cases of the model. As we have seen in the motivation example (Section 1), the proposed model aims at easing the decision process for energy-efficient VM management systems. The estimation of the energy consumption for the different options offered by the system gives information to conduct a “what-if” analysis.

Experimental setup. We conduct experiments with the settings already described in Section 3 with the *ordering* profile workload.

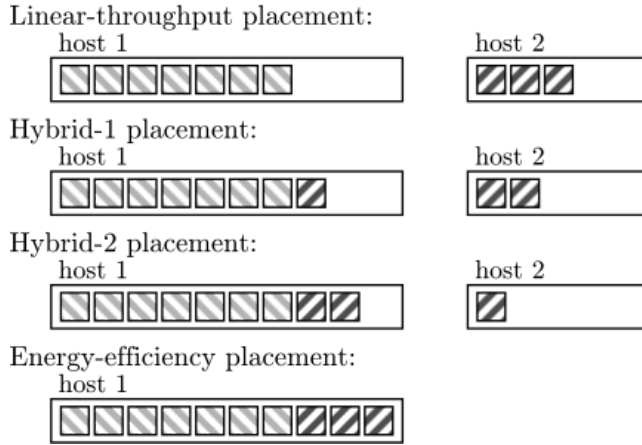
5.1. First use-case: VM allocation

An efficient VM allocation depends on the energy consumed by the host and on the throughput produced by the VMs. We have seen in Section 3.1 that a

small throughput degradation can achieve a high energy savings. In this case, we use the model to estimate the energy consumption of hosts in order to decide for the most suitable VM allocation.



(a) Throughput pattern for the first use-case



(b) Different VMs placements for the first use-case

Figure 13: Number of VMs required and throughput produced in a real execution.

We mimic a system having a workload peak which requires a variable number of VMs. Figure 13a shows the number of VMs required and the throughput produced in a real execution.

The 10 VMs required by the workload can be allocated in one host or can use two hosts. We exploit results from Section 3.1, to evaluate the four VMs placements described in Figure 13b. We have observed that throughput scales linearly until 7 VMs in the same host. Hence, to maintain linear throughput, the next 3 VMs are placed in another host (in the top of the figure). On the other hand, placing 10 VMs in the same host exhibits the largest energy savings with some throughput degradation (in the bottom of the figure). As intermediate

placement, between having linear throughput and having the largest energy savings, we propose two hybrid placements (in the middle of the figure).

We assume that the exact moments to start up and delete the VMs are known. This is not a major constraint in model validation, because the model’s goal is to predict energy consumption of VM management operations and not predicting the actual workload.

Figure 14 presents the results over 10 repetitions. For each possible placement, the figure shows the box plot of the relative error between the value estimated by the model and the real value. The model estimates the energy with a relative error lower than 4% (with average 1.50%). This experiment demonstrates the ability of the proposed model to accurately estimate the energy consumption of systems with peak loads.

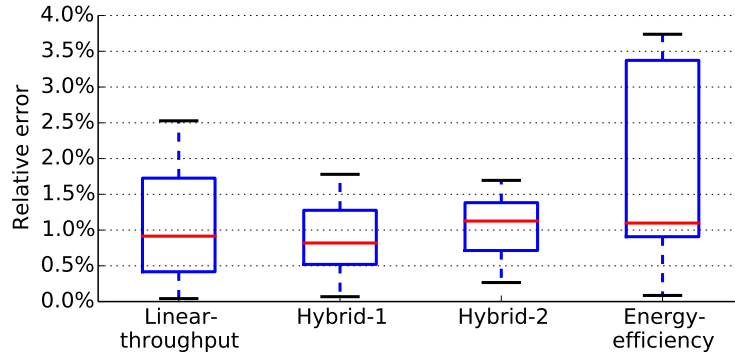


Figure 14: Relative error in the first use-case is lower than 4%.

5.2. Second use-case: starts and migrations together

VMs start-ups and migrations can happen together in the system and the model must be able to fit well also during concurrent operations.

For each host, every 300 seconds one VM management operation starts according to the scenario displayed on Figure 15. For example, for the host 1 allocating 1 initial VM: firstly, a VM starts at 300 seconds (it has 2 VMs); then a VM is received from server 2 at 600 seconds (it has 3 VMs); and finally, another VM migrated from server 3 arrives at 900 seconds (it has 4 VMs). The experiment finishes at 1200 seconds.

Figure 16 presents the results over 10 repetitions. For each host, the figure shows the box plot of the relative error between the value estimated by the model and the real value. The model estimates the energy with a relative error lower than 7% (2.46% on average). This experiment shows that our model correctly grasps the energy consumption of Cloud systems where VM start-up and migrations happen together.

From the achieved results, we show that the proposed model performs well and is able to accurately estimate the energy consumption of VM management

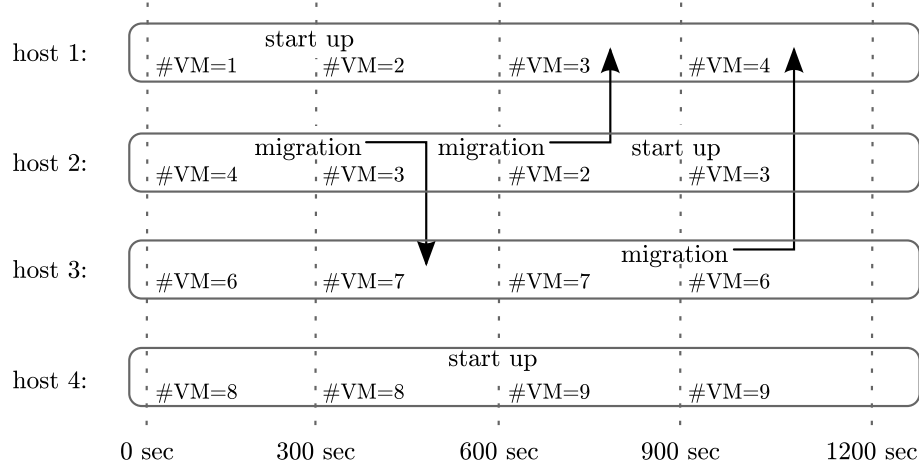


Figure 15: Start-up and migration events together in the second use-case.

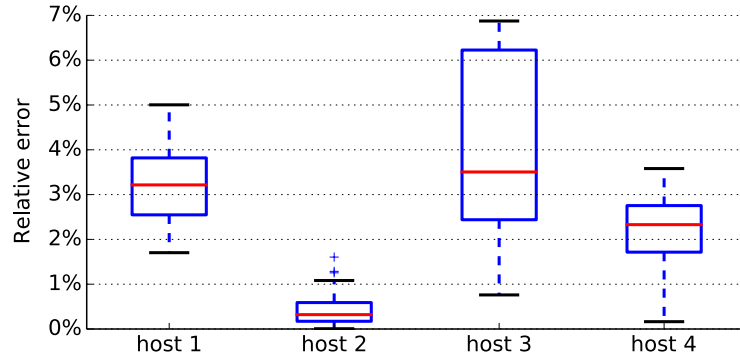


Figure 16: Relative error in the second use-case is lower than 7%.

operations, even when different operations occur at the same time. In all cases, the obtained relative error between estimated and real values are below the 7% line, which is considered as a reasonable accuracy in literature (Section 2).

6. Learning model parameters in a real system

As shown in [11], the instantiation of an energy model is not only influenced by CPU usage, it is application dependent. This means that the estimation of model parameters must be done with the application which is intended to run on the system. In this section, we study how to learn energy parameters for VM consolidation model from previous information. We present how to learn the model parameters from a realistic *training* scenario and we employ another *test* scenario to evaluate the model accuracy over the same application.

Training scenario To learn the model, we start with three hosts allocating eight VMs as shown in Figure 17. Consolidation consists in migrating VMs running on hosts 2 and 3 to host 1. This means, after consolidation hosts 2 and 3 will be completely free. From this scenario, we infer the average duration and power consumption of migration in source and destination hosts.

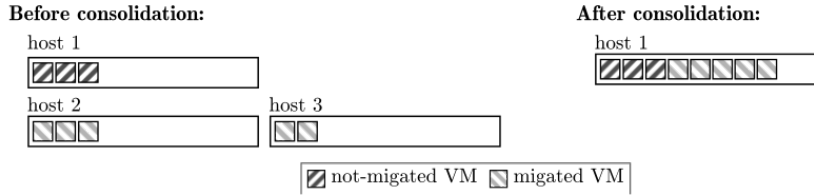


Figure 17: Training scenario for consolidation.

Test scenario In the training scenario, there are not much more choices about how to perform consolidations. However, in a system with more hosts, the number of possible options to perform consolidation (and thus consolidation algorithms) is typically higher – especially if hosts are heterogeneous. As we have seen in the motivation example (Section 1), each algorithm takes different migration choices implying different power consumption values.

In the test scenario, we have four hosts with twelve VMs overall as shown in Figure 18 on the top. In order to have two provisioned hosts after consolidation, algorithms move VMs from hosts 3 and 4 to hosts 1 and 2. Figure 18 on the bottom shows the final configurations for three consolidation algorithms called *first-fit*, *balancing* and *hybrid*.

Results The training set runs 5 times and the test set runs 15 times in order to obtain statistically meaningful results. Figure 19 shows the relative error in the energy estimation over the test set. For each consolidation algorithm, the figure shows the box plot of the relative error between the energy estimated by the model and the real consumption.

The average error is always lower than 10%. However, some error points are greater than 10%. Table 6 shows the percent of experiments having an error

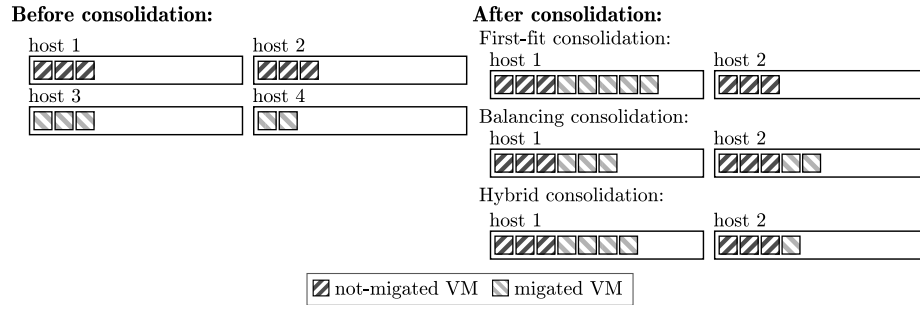


Figure 18: Test scenario for consolidation.

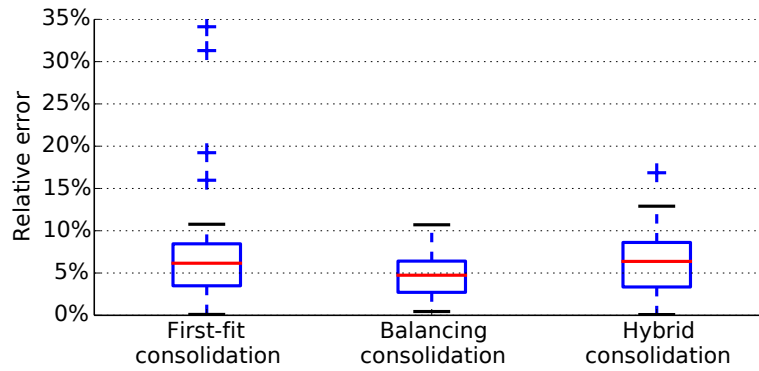


Figure 19: Average relative error is lower than 10% in test scenario.

Table 6: Percent of runs having a relative error greater than 10% in test scenario.

First-fit consolidation	Balancing consolidation	Hybrid consolidation
17.24%	2.74%	14.71%

greater than 10%. This case occurs when the time actually spent in migration is larger than the estimated time. Although the power consumption of the migration is correctly estimated, increasing the time spent in migration produces a larger error.

7. Discussion

We have proposed an experiment-driven model to estimate the energy consumption associated to VM management operations. This model provides reliable and accurate energy consumption estimates required by energy-efficient VM management frameworks.

The proposed model is intended for online applications hosted by web servers where levels of resources utilization are fairly stable. For this reason, our model do not consider performance monitoring counters. In addition, the model does not cover the energy consumption on the network during migration, as done in [42] for instance. However, if one wants to build a complex model targeting different workloads and network consumption, it must be designed based on the principles settled in our model. On top of this solid basis, the next refinement of our model will consist in taking into account VM sizes.

In the experimental facts, energy consumption is confronted to throughput to better understand the energy-performance trade-off. Nevertheless, the presented model only addresses energy aspects. After the experimental investigations presented here, we believe an analogous model can be designed for performance, and it will be the target of a future work.

In addition to the “what-if” analysis enabled by the proposed model, the energy consumption dependence on the VM management operations shows a potential to increase energy savings in Cloud platforms. Energy savings could come from the opportunity to exchange resource-level and application-level information. While Infrastructure-as-a-Service (IaaS) energy-savings approaches actually provide energy savings, they can only base their decision on low-level metrics such as CPU usage, memory, I/O etc. However, these infrastructure-level metrics are insufficient to give enough insight on the behavior of applications that generate the workload. Instead of treating applications as black boxes, we argue that significant energy savings could result from allowing the consolidation algorithms to access application-level knowledge available at the Platform-as-a-Service (PaaS) level of the Cloud stack. For example, only the PaaS layer can know if the workload is expected to decrease in the next few minutes, thus provoking the halt of some existing VMs. Making this information

available in the consolidation system would allow to avoid costly and unnecessary actions such as migrating a VM while the PaaS plans to stop it soon. Such energy-related decisions could be taken by the IaaS using our model to perform the “what-if” analysis including PaaS information, and could thus lead to cooperative energy saving strategies for Cloud platforms.

8. Conclusions

In this paper, we investigated the energy consumption associated to VM management operations such as VM placement, VM start up and VM migration. Firstly, we studied, through an extensive set of experiments, the performance and energy consumption, the impact of the different operations and the role of the number of running VMs on the considered hosts. Secondly, we propose a model for estimating the energy consumption with an average error lower than 10% for transactional web applications allowing overcommit situations. Such an average error is similar to the one obtained in literature for estimating the energy consumption of a Cloud server [31, 27, 32, 30, 28, 29] and the energy consumption of VM migration [37, 39, 42]. Unlike previous work, this model shows the relation between the energy consumption induced by a single VM and server configuration in terms of hosted VMs. In addition, to the best of our knowledge, our energy model for VM management systems is the first to work in the case of overcommit situations, that are frequent in real Cloud environment.

As future work, we will extend the evaluation to a wide range of applications, such as data intensive applications. Furthermore, we will introduce the throughput behavior in the model to estimate the throughput-energy trade-off providing a whole picture of the system.

Acknowledgments

This work was partially funded by the EcoPaaS project as part of the “Stratégie d’attractivité durable” program of the Brittany regional council. Experiments presented in this paper were carried out using the Grid’5000 experimental test-bed, being developed under the INRIA ALADDIN development action with support from CNRS, RENATER and several Universities as well as other funding bodies (see <https://www.grid5000.fr>).

References

- [1] J. Koomey, Growth in data center electricity use 2005 to 2010, Analytics Press.
- [2] Z. A. Mann, Rigorous results on the effectiveness of some heuristics for the consolidation of virtual machines in a cloud data center, *Future Generation Computer Systems* 51 (2015) 1 – 6.

- [3] A.-C. Orgerie, M. D. de Assuncao, L. Lefèvre, A survey on techniques for improving the energy efficiency of large-scale distributed systems, *ACM Computing Surveys* 46 (4).
- [4] TPC Benchmark W (Web Commerce) Specification. version 1.0.1, www.tpc.org/tpcw (2000).
- [5] S. A. Baset, L. Wang, C. Tang, Towards an Understanding of Oversubscription in Cloud, in: *USENIX Workshop on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services (Hot-ICE)*, USENIX, 2012.
- [6] D. Breitgand, Z. Dubitzky, A. Epstein, A. Glikson, I. Shapira, SLA-aware Resource Over-commit in an IaaS Cloud, in: *International Conference on Network and Service Management (CNSM)*, International Federation for Information Processing, 2013, pp. 73–81.
- [7] C. Mobius, W. Dargie, A. Schill, Power Consumption Estimation Models for Processors, Virtual Machines, and Servers, *IEEE Transactions on Parallel and Distributed Systems* 25 (6) (2014) 1600–1614.
- [8] A. Jaialtilal, Y. Jiang, S. Mishra, Modeling CPU Energy Consumption for Energy Efficient Scheduling, in: *Workshop on Green Computing (GCM)*, 2010, pp. 10–15.
- [9] M. Maggio, H. Hoffmann, A. V. Papadopoulos, J. Panerati, M. D. Santambrogio, A. Agarwal, A. Leva, Comparison of Decision-Making Strategies for Self-Optimization in Autonomic Computing Systems, *ACM Transactions on Autonomous and Adaptive Systems* 7 (4) (2012) 36:1–36:32.
- [10] N. Mishra, H. Zhang, J. D. Lafferty, H. Hoffmann, A Probabilistic Graphical Model-based Approach for Minimizing Energy Under Performance Constraints, in: *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2015, pp. 267–281.
- [11] A. Orgerie, L. Lefèvre, J. Gelas, Demystifying energy consumption in Grids and Clouds, in: *International Green Computing Conference 2010, IGCC '10*, 2010, pp. 335–342.
- [12] P. Petrica, A. M. Izraelevitz, D. H. Albonesi, C. A. Shoemaker, Flicker: A Dynamically Adaptive Architecture for Power Limited Multicore Systems, *ACM SIGARCH Computer Architecture News* 41 (3) (2013) 13–23.
- [13] S. Sridharan, G. Gupta, G. S. Sohi, Holistic Run-time Parallelism Management for Time and Energy Efficiency, in: *International ACM Conference on International Conference on Supercomputing (ICS)*, 2013, pp. 337–348.

- [14] A. Gandhi, V. Gupta, M. Harchol-Balter, M. A. Kozuch, Optimality analysis of energy-performance trade-off for server farm management, *Performance Evaluation* 67 (11) (2010) 1155–1171.
- [15] D. Liao, K. Li, G. Sun, V. Anand, Y. Gong, Z. Tan, Energy and performance management in large data centers: A queuing theory perspective, in: *International Conference on Computing, Networking and Communications, ICNC '15*, 2015, pp. 287–291.
- [16] A. Beloglazov, J. H. Abawajy, R. Buyya, Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing, *Future Generation Computer Systems* 28 (5) (2012) 755–768.
- [17] A. Beloglazov, R. Buyya, Energy Efficient Allocation of Virtual Machines in Cloud Data Centers, in: *10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, CCGrid '10*, 2010, pp. 577–578.
- [18] D. Borgetto, H. Casanova, G. D. Costa, J. Pierson, Energy-aware service allocation, *Future Generation Computer Systems* 28 (5) (2012) 769–779.
- [19] H. Li, G. Zhu, C. Cui, H. Tang, Y. Dou, C. He, Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing, *Computing* (2015) 1–15.
- [20] D. Barbagallo, E. D. Nitto, D. J. Dubois, R. Mirandola, A Bio-inspired Algorithm for Energy Optimization in a Self-organizing Data Center, in: *First International Workshop on Self-Organizing Architectures, SOAR '09*, 2009, pp. 127–151.
- [21] M. Mazzucco, D. Dyachuk, R. Deters, Maximizing cloud providers' revenues via energy aware allocation policies, in: *IEEE International Conference on Cloud Computing, CLOUD '10*, 2010, pp. 131–138.
- [22] T. Mastelic, A. Oleksiak, H. Claussen, I. Brandic, J.-M. Pierson, A. V. Vasilakos, Cloud Computing: Survey on Energy Efficiency, *ACM Computer Surveys* 47 (2) (2014) 33:1–33:36.
- [23] A. Strunk, W. Dargie, Does Live Migration of Virtual Machines Cost Energy?, in: *IEEE International Conference on Advanced Information Networking and Applications (AINA)*, 2013, pp. 514–521.
- [24] A. Verma, G. Kumar, R. Koller, A. Sen, CosMig: Modeling the Impact of Reconfiguration in a Cloud, in: *International Symposium on Modeling, Analysis Simulation of Computer and Telecommunication Systems (MAS-COTS)*, 2011, pp. 3–11.
- [25] K. Ye, X. Jiang, D. Huang, J. Chen, B. Wang, Live Migration of Multiple Virtual Machines with Resource Reservation in Cloud Computing Environments, in: *IEEE International Conference on Cloud Computing (CLOUD)*, 2011, pp. 267–274.

- [26] OpenStack Project, OpenStack: The open source cloud operating system, openstack.org.
- [27] A. E. H. Bohra, V. Chaudhary, VMeter: Power modelling for virtualized clouds, in: 24th IEEE International Symposium on Parallel and Distributed Processing - Workshop Proceedings, IPDPS '10, 2010, pp. 1–8.
- [28] R. Bertran, Y. Becerra, D. Carrera, V. Beltran, M. González, X. Martorell, N. Navarro, J. Torres, E. Ayguadé, Energy accounting for shared virtualized environments under DVFS using PMC-based power models, *Future Generation Computer Systems* 28 (2) (2012) 457–468.
- [29] M. Colmant, M. Kurpicz, P. Felber, L. Huertas, R. Rouvoy, A. Sobe, Process-level power estimation in VM-based systems, in: European Conference on Computer Systems (EuroSys), 2015, p. 14.
- [30] G. Dhiman, K. Mihic, T. Rosing, A system for online power prediction in virtualized environments using Gaussian mixture models, in: Design Automation Conference (DAC), 2010, pp. 807–812.
- [31] A. Kansal, F. Zhao, J. Liu, N. Kothari, A. A. Bhattacharya, Virtual Machine Power Metering and Provisioning, in: ACM Symposium on Cloud Computing (SoCC), 2010, pp. 39–50.
- [32] B. Krishnan, H. Amur, A. Gavrilovska, K. Schwan, VM power metering: feasibility and challenges, *SIGMETRICS Performance Evaluation Review* 38 (3) (2010) 56–60.
- [33] A. Carpen-Amarie, A. Orgerie, C. Morin, Experimental Study on the Energy Consumption in IaaS Cloud Environments, in: IEEE/ACM 6th International Conference on Utility and Cloud Computing, UCC '13, 2013, pp. 42–49.
- [34] S. B. S. P. E. Corporation, www.spec.org/jbb2013/.
- [35] Q. Huang, F. Gao, R. Wang, Z. Qi, Power Consumption of Virtual Machine Live Migration in Clouds, in: Third International Conference on Communications and Mobile Computing, CMC '11, 2011, pp. 122–125.
- [36] A. Strunk, Costs of Virtual Machine Live Migration: A Survey, in: Eighth IEEE World Congress on Services, SERVICES '12, 2012, pp. 323–329.
- [37] H. Liu, C. Xu, H. Jin, J. Gong, X. Liao, Performance and energy modeling for live migration of virtual machines, in: ACM International Symposium on High Performance Distributed Computing (HPDC), 2011, pp. 171–182.
- [38] Xen, <http://www.xenproject.org>.
- [39] A. Strunk, A lightweight model for estimating energy cost of live migration of virtual machines, in: 2013 IEEE Sixth International Conference on Cloud Computing, CLOUD '13, 2013, pp. 510–517.

- [40] KVM: Kernel-based Virtual Machine, <http://www.linux-kvm.org>.
- [41] V. D. Maio, G. Kecskemeti, R. Prodan, Workload-Aware Energy Model for Virtual Machine Migration, in: IEEE International Conference on Cluster Computing, CLUSTER '15, 2015.
- [42] V. D. Maio, R. Prodan, S. Benedict, G. Kecskemeti, Modelling energy consumption of network transfers and virtual machine migration, Future Generation Computer Systems.
- [43] D. Balouek, A. Carpen Amarie, G. Charrier, F. Desprez, E. Jeannot, E. Jeanvoine, A. Lèbre, D. Margery, N. Niclausse, L. Nussbaum, O. Richard, C. Pérez, F. Quesnel, C. Rohr, L. Sarzyniec, Adding virtualization capabilities to the Grid'5000 testbed, in: I. Ivanov, M. Sinderen, F. Leymann, T. Shan (Eds.), Cloud Computing and Services Science, Vol. 367 of Communications in Computer and Information Science, Springer International Publishing, 2013, pp. 3–20.
- [44] A. E. Instances, https://aws.amazon.com/ec2/instance-types/?nc1=h_ls.
- [45] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, A. Warfield, Live Migration of Virtual Machines, in: Conference on Symposium on Networked Systems Design & Implementation (NSDI), 2005, pp. 273–286.
- [46] P. Svärd, B. Hudzia, S. Walsh, J. Tordsson, E. Elmroth, Principles and Performance Characteristics of Algorithms for Live VM Migration, SIGOPS Operating Systems Review 49 (1) (2015) 142–155.
- [47] Nvidia Corporation, www.nvidia.com/object/what-is-gpu-computing.html#sthash.TDnNYEDE.dpuf.



Mar Callau-Zori received her PhD. degree in Computer Science from Technical University of Madrid (Spain) in January 2013. She was working as a postdoc at the LIP6 at the Pierre and Marie Curie University (France) between 2013–2014 on the MyCloud project. From May 2014 to November 2015, she was working as a postdoc at IRISA laboratory (Rennes, France). Her current research interests involve data stream mining, energy efficiency and algorithms for distributed systems.



Lavinia Samoila has received her Master's degree in Computer Science from the University of Rennes 1, France and the Royal Institute of Technology, Sweden. During her studies, she has been an intern at IRISA, assisting with setting up the test environments. She is currently a Data Engineer at Spotify, working with improving the data pipelines. Her main interests are large data processing frameworks, machine learning algorithms and computer networking.



Anne-Cécile Orgerie received her PhD. degree in Computer Science from École Normale Supérieure de Lyon (France) in September 2011. From October 2011 to September 2012, she was working as a postdoc at the Department of Electrical and Electronic Engineering at the University of Melbourne (Australia) on the PetaFlow project (French-Japanese project). She has been a full time researcher at CNRS in the IRISA laboratory (Rennes, France) since October 2012. Her research interests focus on energy efficiency, cloud computing, and distributed systems.



Guillaume Pierre is a Professor in Computer Science at the University of Rennes 1, France. Prior to this he spent 13 years at the VU University Amsterdam. His main interests are Cloud computing, Web application support, peer-to-peer and many other types of large-scale distributed systems. He took part in several European and EIT ICT Labs projects and acted as the lead designer of the ConPaaS platform-as-a-Service environment. Pierre holds a PhD in Computer Science from the University of Evry-val d'Essonne, France. He is also the academic coordinator of the EIT ICT Labs Master school at the University of Rennes 1.